8 Using Hardware Manager

Overview

This section provides information about the Hardware Manager, a DLL (hwmgr.dll) which provides the following capabilities:

- maintains an ASCII text Hardware Setup file, which contains detailed, specific information about one or more DSP/analog hardware types
- maintains mapping of logical "hardware variables" to currently specified physical "hardware values" for use by other Signalogic DSP Software components
- provides access to the Hardware Setup File, and to current logical hardware variables and physical hardware values through an API
- generates and displays on demand the Hardware Selector dialog box, which allows DSP/analog hardware to be selected from a list box and configured by the user

The Hardware Manager DLL is included with DSPower-HWLib and DSPower-Block Diagram software. However, most of the flexibility and customization features of the Hardware Manager arise from its use with DSPower-HWLib software. For example, the Hardware Setup

file can easily be customized to display only certain hardware types, or to application-specific hardware descriptions, using a standard text editor. See section 8.1.2 below, Hardware Setup File.

For information about DSPower-HWLib software, see chapter 4, Using DSPower-HWLib and DSPower-SPLib Software.

Main Topics

Hardware Manager Access

Hardware Selector

Selecting DSP/Analog Hardware Types

DSP/Analog Hardware Configuration

Hardware Values and Variables

Hardware Setup File

Error Handling in Hardware Manager

8.1 Hardware Manager Access

[@k implicit loading, explicit loading]

To access Hardware Manager, the hwmgr.dll library must be visible to your application's calling program. The easiest way to do this is to include the hwmgr.lib file in your project. For C/C++ projects, please see the example Borland C/C++ v3.1 and v4.5, and the Microsoft Visual C/C++ v1.5 and v2.0, project files, included with DSPower-HWLib software. For MATLAB projects, please see the example .m files. Once included in your project file, and thus linked into your final executable, the enmgr.lib file causes the Engine Manager DLL to be loaded and initialized when your executable first runs (actually, *before* any code in the executable runs). This is known as implicit loading.

Another way of accessing the hwmgr.dll library is to load it from your application program dynamically, using the LoadLibrary call in the Windows API, followed by one or more GetProcAddress calls with specific function names as input parameters. This method is known as

explicit loading, and is more complicated, requiring painstaking typecasting to create function prototype *pointers* which are set as a result of GetProcAddress calls. "Behind the scenes", this is also the technique used by MATLAB to load DLLs needed by calling .m programs--except that MATLAB simplifies the process by allowing DLL calls to look much like other MATLAB function calls.

Also, for C/C++ programs, the hwmgr.h include file must be specified along with other include files that may be needed by the calling application. Note that the hwmgr.h include file expects the standard Windows 3.1 windows.h include file to have been previously specified.

NOTE: For C/C++ development, make sure the include file directory list in your IDE contains the DSPower-HWLib subdirectory. The example project files supplied with DSPower-HWLib software assume default include file subdirectories of "hwlib", "toollib", and "helplib" under the root-level installation subdirectory, although toollib and helplib are present only for demo purposes.

General Topics

Hardware Selector

Selecting DSP/Analog Hardware Types

DSP/Analog Hardware Configuration

Hardware Values and Variables

Hardware Setup File

Error Handling in Hardware Manager

8.2 Hardware Selector

In DSPower-Block Diagram software, the Hardware Selector is activated by selecting Configuration | DSP/Analog Hardware. The Hardware Selector dialog box should then appear.

When using DSPower-HWLib software, the Hardware Selector is activated when a calling program makes the DSShowHardwareSelectorDlg call into hwmgr.dll. The prototype for this call is made available by including the hwmgr.h file.

Upon activation, the Hardware Selector dialog box displays the following:

- A hardware type listbox, which lists possible hardware selections alphabetically by manufacturer.
- A hardware configuration section, which allows configuration of important hardware parameters, including I/O and/or memory base addresses, host PC bus type, default executable DSP file, processor speed, and more. The current contents of the hardware configuration section reflect the current hardware type listbox selection, and change when the selected item in the listbox changes.
- A "hardware variable" section, which allows mapping of hardware variables commonly used in DSPower-Block Diagram software functions to different hardware types.

Note that the selections available in the hardware type listbox, and thus what can appear in the hardware configuration section, is entirely determined by the current contents of the Hardware Setup file (hwsetup.lst), which is maintained by the Hardware Manager DLL (hwmgr.dll). See section 8.3 below, Hardware Setup File, for more information.

Related Topics

General Topics

Hardware Manager Access

Hardware Selector

Hardware Setup File

Error Handling in Hardware Manager

8.2.1 Selecting DSP/Analog Hardware Types

After activating the Hardware Selector (see section 8.2 above, Using the Hardware Selector), a dialog box is displayed that includes a listbox showing possible hardware selections. The contents of the listbox select the current contents of the Hardware Setup file (hwsetup.lst). In the Hardware Setup file, hardware entries are organized in groups by manufacturer; in the listbox, the manufacturer name is prefixed to the hardware type, causing all hardware selections to be listed alphabetically by manufacturer.

When selecting a hardware type, note that there can be one or more closely related selections for the same basic hardware. For example, the Ariel PC-56D board can be selected to use its onboard analog I/O, or for use with Ariel's Digital Microphone, ProPort, or DatPort. Thus, keep the following in mind:

1. If you don't see your selection right away, make sure you check all possibilities for that manufacturer.
2. Check closely to make sure you have the right selection--even a small mistake can make the difference between a hardware selection working correctly or not.

After highlighting (selecting) the desired hardware type, you should see its corresponding configuration parameters displayed in the hardware configuration section of the Hardware Selector dialog box. This is the area directly below the hardware type listbox. For a rundown of these options, see section 8.2.2 below, DSP/Analog Hardware Configuration.

Related Topics

DSP/Analog Hardware Configuration

Hardware Values and Variables

General Topics

Hardware Manager Access

Hardware Selector

Hardware Setup File

Error Handling in Hardware Manager

8.2.2 DSP/Analog Hardware Configuration

When the Hardware Selector dialog box is displayed, it can be divided into three major areas:

- A hardware type listbox, which lists possible hardware selections alphabetically by manufacturer. See section 8.2.1 above, Selecting DSP/Analog Hardware Types.
- A hardware configuration section (underneath the hardware type listbox), which allows configuration of important hardware parameters, including I/O and/or memory base addresses, host PC bus type, default executable DSP file, processor speed, and more. See the description of selectable items below.
- *For DSPower-Block Diagram software only,* an additional "hardware variable" dialog box, which can be invoked using the "Set Hardware Variables" button. This dialog box allows mapping of hardware variables commonly used in DSPower-Block Diagram software functions to different hardware values. See section 8.2.3 below, Hardware Variables and Values.

Whenever parameters are changed in the hardware type listbox or hardware configuration sections of the Hardware Selector and the OK button is clicked, changes are saved to the Hardware Configuration file (see section 8.3, Hardware Configuration File). Changes are discarded if the Cancel button is clicked.

See below for descriptions of the fields inside the hardware configuration section of the Hardware Selector.

Hardware Configuration Parameters

Board Designator

I/O Base Address

Memory Base Address

Processor Clock

Voltage Ranges

Number of Processors

DSP Program File

Analog Clock Type

Bus Type

Memory Architecture

Related Topics

Selecting DSP/Analog Hardware Types

Hardware Values and Variables

General Topics

Hardware Manager Access

Hardware Selector

Hardware Setup File

Error Handling in Hardware Manager

Board Designator

[@j board designator string]

Board designators are short string descriptions specifying hardware types, typically comprised of a manufacturer abbreviation and DSP device abbreviation. At any time, the designator shown reflects the DSP/analog hardware type currently selected. Board designators cannot be edited directly in the Hardware Selector; modification, if necessary, can be achieved by modifying the Hardware Setup File using the DSSetHWMgrEntry function or directly editing it. Note, however, that changing board designators or adding new ones is not normally recommended. See Board Designator Field in Hardware Setup File, in section 8.3.1 below, Hardware Setup File Fields.

Hardware Configuration Parameters

Board Designator
I/O Base Address
Memory Base Address
Processor Clock
Voltage Ranges
Number of Processors
DSP Program File
Analog Clock Type
Bus Type
Memory Architecture

I/O Base Address

For DSP/analog hardware with host PC interfaces using an I/O-mapped addressing scheme, this field should be set to match the base I/O address established by the jumpers or switch(es) on the hardware. Typical examples are 300H, 290H, 340H, etc. Values should include the 'H' suffix if entered in hexadecimal; values can also be entered in "C" hexadecimal notation (e.g. 0x300, 0x290, 0x340).

Note that some hardware do not actually include jumpers or switches; but in these cases, selecting an appropriate I/O base address value still needs to be done correctly.

Nearly all hardware types use an I/O base address even if a memory base address. Typically, there may actually be from 8 to 32 individual I/O locations offset from the base address. If a Hypersignal-Macro Series software package is being used as a DSP Engine, see chapter 3 in the Hypersignal-Macro Series Hardware Reference Guide for information about I/O base addresses for specific hardware types.

For more information about I/O address conflicts which can arise with other installed hardware in the host PC, see Signalogic DSP Software System Guide section 4.3, Troubleshooting DSP/Analog Hardware Problems.

Hardware Configuration Parameters

Board Designator

I/O Base Address

Memory Base Address

Processor Clock

Voltage Ranges

Number of Processors

DSP Program File

Analog Clock Type

Bus Type

Memory Architecture

Memory Base Address

For DSP/analog hardware with a host PC interface using an memory-mapped addressing scheme, this field should be set to match the base memory address established by the jumpers or switch(es) on the hardware. Typical examples are D000H, DC00H, DE00H, etc. Values should include the 'H' suffix if entered in hexadecimal; values can also be entered in "C" hexadecimal notation (e.g. 0xd000, 0xdc00, 0xde00).

Note that some hardware do not actually include jumpers or switches; but in these cases, selecting an appropriate memory base address value still needs to be done correctly.

Some, but not all, hardware types use a memory base address even if an I/O base address (see I/O Base Address, above) is also used. If a memory base address does apply, then the actual size of the memory segment used in the host PC's memory space may be from 2k to 64k bytes. If a Hypersignal-Macro Series software package is being used, see chapter 3 in the Hypersignal-Macro Series Hardware Reference Guide for information about memory base addresses for specific hardware types.

NOTE: If a memory base address is used, your system almost certainly will need the memory-mapped area to be "reserved" using the EMM386.EXE line in the system's config.sys file. This is necessary to

avoid conflicts with memory-resident software programs and utilities, as well as other installed hardware. For more information about this, see Signalogic DSP Software System Guide section 1.4.4, System Drivers and Resident Program Considerations, and EMM386.EXE Considerations (Reserving Memory Areas) in Signalogic DSP Software System Guide section 4.3.2, Memory-Mapped Transfer Gotchas.

Hardware Configuration Parameters

Board Designator

I/O Base Address

Memory Base Address

Processor Clock

Voltage Ranges

Number of Processors

DSP Program File

Analog Clock Type

Bus Type

Memory Architecture

Processor Clock

The Processor Clock field specifies the rate, in MHz, of the clock signal supplied to the DSP device on the selected DSP/analog hardware. This value is used by DSPower series and Hypersignal-Macro series software to calculate sampling frequency settings in cases where the DSP/analog hardware derives its sampling rate values from the processor clock.

NOTE: In some cases, the processor's speed may not be the same as its clock rate. For example, DSP manufacturers may talk about an "100 MFLOPs device", which uses a 40 MHz clock input. Or they speak of a 40 MHz device, which actually takes a 20 MHz clock.

Hardware Configuration Parameters

Board Designator
I/O Base Address
Memory Base Address
Processor Clock
Voltage Ranges
Number of Processors
DSP Program File
Analog Clock Type
Bus Type
Memory Architecture

Voltage Ranges

The Voltage Ranges field sets the analog input and output voltage ranges for the selected DSP/analog hardware type. Entry is assumed to be in Volts and should be of the form:

```
inp,out
```

where inp and out are floating point values typically ranging from 0 to 10. For example, an entry of:

```
1.8,3.0
```

would specify a ±1.8V input voltage range, and a ±3V output voltage range. Note that this parameter is significant only for hardware which has (software-selectable) programmable gain. In such cases, gain can be determined by comparing the entered value to that made in individual instrument and data acquisition functions. Otherwise, when hardware gain cannot be controlled via software, voltage range entry made in individual instrument and data acquisition functions have no effect on signal gain. In these cases, voltage ranges should exactly match the actual hardware settings, in order to produce correct amplitude units display. See chapter 3 in the Hypersignal-Macro Series Hardware Reference Guide for voltage range specifications for individual hardware types.

Hardware Configuration Parameters

Board Designator

I/O Base Address

Memory Base Address

Processor Clock

Voltage Ranges

Number of Processors

DSP Program File

Analog Clock Type

Bus Type

Memory Architecture

Number of Processors

The Number of Processors field reflects the number of processors on the DSP/analog hardware type currently selected. This field cannot be changed in the Hardware Selector; if modification is necessary, the Hardware Setup File must be edited directly. See Number of Processors Field in Hardware Setup File, in section 8.3.1 below, Hardware Setup File Fields.

Hardware Configuration Parameters

Board Designator

I/O Base Address

Memory Base Address

Processor Clock

Voltage Ranges

Number of Processors

DSP Program File

Analog Clock Type

Bus Type

Memory Architecture

DSP Program File

The DSP Program File field specifies the default executable DSP file to download to the DSP/analog hardware if a file is not explicitly specified in a function, display, or instrument block in DSPower-Block Diagram software, or in DSLoadFileBoard and DSLoadFileProcessor calls in Hardware Library (included with DSPower-HWLib software).

The DSP Program File field should specify a file fully prepared for run-time initialization and execution, in the output format used by the DSP device manufacturer's linker utility. Typically, these linkers produce COFF (common object file format) files, which are handled automatically by the DSLoadxxx calls in Hardware Library.

The default filenames for this field are set to the files shipped standard with Hypersignal-Macro series software. For information about creating and modifying DSP executable files for use with Hypersignal-Macro Series software, see chapter 5, Using DSP Source Code Interface Software.

Hardware Configuration Parameters

Board Designator

I/O Base Address

Memory Base Address

Processor Clock

Voltage Ranges

Number of Processors

DSP Program File

Analog Clock Type

Bus Type

Memory Architecture

Analog Clock Type

The Analog Clock Type selection specifies the type of analog I/O clock (sampling frequency clock) used on the DSP/analog hardware. This selection should be one of the following:

none

external

fixed, generated onboard or onchip

software programmable, generated onboard or onchip

NOTE: If the Analog Clock Type field is set to external or fixed internal, then you must remember to accurately enter sampling rate values in data acquisition, display, and instrument functions to exactly match the fixed value. Otherwise, axis and units readouts may be incorrect.

Hardware Configuration Parameters

Board Designator

I/O Base Address

Memory Base Address

Processor Clock

Voltage Ranges

Number of Processors

DSP Program File

Analog Clock Type

Bus Type

Memory Architecture

Bus Type

The Bus Type selection specifies the host computer bus type for the hardware. For most DSP/analog hardware, the "80286" or similar button should be selected, indicating that the bus interface on the

hardware can handle 80286-type string move instructions at a minimum. Below are other Bus Type selections:

8086 instruction set only

80286 instruction set (string move instructions)

(future selections reserved for Signalogic)

Hardware Configuration Parameters

Board Designator

I/O Base Address

Memory Base Address

Processor Clock

Voltage Ranges

Number of Processors

DSP Program File

Analog Clock Type

Bus Type

Memory Architecture

Memory Architecture

The Memory Architecture field reflects the memory architecture of the DSP/analog hardware type currently selected. This field cannot be changed in the Hardware Selector; if modification is necessary, the Hardware Setup File must be edited directly. See Memory Architecture Field in Hardware Setup File, in section 8.3.1 below, Hardware Setup File Fields.

The Memory Architecture field applies to onboard memory that is external to the DSP device on the DSP/analog hardware, and not necessarily to onchip memory. This field should indicate one of the following architecture types:

Lin	**linear**, both program and data in the same memory space
Harv	modified **Harvard** architecture, with program and data linear, but contained in separate memory spaces

Vec modified Harvard architecture with program and data in separate memory spaces, and with data memory further subdivided into **vector** memory spaces

Hardware Configuration Parameters

Board Designator

I/O Base Address

Memory Base Address

Processor Clock

Voltage Ranges

Number of Processors

DSP Program File

Analog Clock Type

Bus Type

Memory Architecture

8.2.3 Hardware Values and Variables

[@j hardware value, hardware variable]

In DSPower series software, a hardware value can be initialized in two ways:

1. when a hardware type is selected in the Hardware Selector and configured as needed
2. when DSAssignBoard calls (and/or other calls) are made to the DSPower-HWLib DLL

There are as many hardware values possible as there are items in the hardware type listbox in the Hardware Setup file.

Hardware variables, on the other hand, are *logical references* to a classification of hardware, and may be one of several different values, and which may change in value dynamically. For example, a function might refer to the current "DSP Accelerator" hardware variable to

attempt to execute all or part of its task on installed DSP hardware. In this manner, the function does not care or know about the specific type of hardware installed, or whether it has changed.

Note that the current hardware value setting and hardware variable settings are not stored in the Hardware Setup File explained in section 8.3 below. They are instead stored in a separate private configuration file maintained by Hardware Manager (hwmgr.ini).

Hardware variables can be assigned using the "Set Hardware Variables" button in the Hardware Selector. Currently, this button is active only in DSPower-Block Diagram software. In DSPower-Block Diagram software, the Hardware Selector appears after selecting Configuration | DSP/Analog Hardware from the top-level worksheet menu. One use of hardware variables in DSPower-Block Diagram software is to allow operation of more than one type of DSP/analog hardware simultaneously when executing block diagrams.

Related Topics

Selecting DSP/Analog Hardware Types

DSP/Analog Hardware Configuration

General Topics

Hardware Manager Access

Hardware Selector

Hardware Setup File

Error Handling in Hardware Manager

8.3 Hardware Setup File

The Hardware Setup File is normally shipped in a default state listing all boards supported by Signalogic DSP Software components. It is an ASCII text file, with no special characters, and is installed by the Signalogic Software Setup Utility under the "xxx\hwlib" subdirectory as "hwsetup.lst", where "xxx" is the subdirectory specified during the DSPower series software installation (for example, "c:\dspower").

Currently, a single Hardware Setup File is maintained by Hardware Manager, regardless of the number of calling applications.

The Hardware Setup File is initially read into global memory from disk file upon the first instance of a call into Hardware Manager requiring data DSP/analog hardware configuration data. This occurs automatically, and includes the DSShowHardwareSelectorDlg call. Global memory is allocated as each new entry is read from the Hardware Setup File. If at some point, memory cannot be increased, NULL is returned and a subsequent call to DSGetHWMgrErrorStatus will return DSCOULDNOTINCREASEHWLISTMEM.

It is possible to change DSP/analog hardware configuration data by calling the DSSetHWMgrEntry function with a pointer to a HWLISTENTRY structure. In this case, a DSWriteHWSetupFile call should be made if DSP/analog hardware configuration data has been changed, and needs to be saved. Note that if the Hardware Selector is invoked, and DSP/analog hardware configuration data is changed by the operator, followed by pushing the "OK" button, DSWriteHWSetupFile is executed automatically. For more information about accessing the Hardware Setup File and changing DSP/analog hardware configuration data, see section 8.3.2 below, Hardware Setup File Access, and HWLISTENTRY Structure in section 8.3.2.

Below is a partial segment of a typical Hardware Setup file, showing three (3) entries, where each entry is a line in the file. Note that the entries are shown with each line divided into two sections, due to the limited width of the printed version of the documentation.

Description	Board Designator	Driver IDs	Num Proc	I/O Base Addr	Mem Base Addr
DSP-16	ARI320	2 0	1 1	33C 33C	D000 D000
DSP-32C	ARI32C	30 0	1 1	300 300	none none
DSP-56	ARI56	29 4	1 1	340 340	none none

Bus Type	Anal Clock Type	Voltage Ranges	DSP Program File	Proc Clock	Mem Arch
1 1	3 3	10,10 10,10	tms32025.out tms32025.out	50 50	L L
0 0	3 3	5.0,5.0 5.0,5.0	dsp32c.out dsp32c.out	50 50	L L

```
1 1     3 3     5.0,5.0     dsp56001.out  dsp56001.out    20  20      V V
                5.0,5.0
```

Notes

1) Except for the board designator and driver ID fields, the second parameter in each field is the default and is not changed by Hardware Manager. If the user selects Restore from Defaults, this parameter is used to re-initialize the field. *The board designator and driver ID fields should not be changed under any circumstances. Doing so will cause undefined results.*

2) All parameters must be separated by at least two (2) spaces to be deemed separate.

3) Addresses are in hex ("0x" prefix or "h" suffix allowed).

4) The default DSP program files are set to executable DSP files shipped standard with Hypersignal-Macro series software, but can be changed as needed. For information about customizing Hypersignal DSP Source Code Interface files, see chapter 5, Using DSP Source Code Interface Software.

5) Board designator strings can be used directly in DSPower-HWLib DSAssignBoard and DSSetHWMgrCurVal calls. The DSShowHardwareSelectorDlg call returns a board designator string.

6) Driver IDs are subdivided into two parts: a major ID, and a minor ID. See section 8.2.3 below, Hardware Setup File: Driver IDs, for more information.

7) Entries in the Hardware Setup file are subdivided by manufacturer section names, such "[Ariel]", or "[Data Translation]", etc. Manufacturer section names occupy a line by themselves, and are used to cause alphabetic display in the hardware type listbox portion of the Hardware Selector dialog box (see section 8.2 above, Hardware Selector).

Whenever parameters are changed in the hardware type listbox or hardware configuration sections of the Hardware Selector (see section 8.2.2, DSP/Analog Hardware Configuration), and the OK button is clicked, changes are saved to the Hardware Configuration file. Changes are discarded if the Cancel button is clicked.

Related Topics

General Topics

Hardware Manager Access

Hardware Selector

Selecting DSP/Analog Hardware Types

DSP/Analog Hardware Configuration

Hardware Values and Variables

Hardware Setup File

Error Handling in Hardware Manager

8.3.1 Hardware Setup File Fields

This section contains a breakdown of individual fields in the Hardware Setup file. Note that these fields are a superset of the fields shown in the Hardware Selector dialog box, including all fields described in section 8.2.2 above, DSP/Analog Hardware Configuration. However, the Hardware Setup file includes several fields not displayed in the Hardware Selector dialog box. These are fields which cannot be changed dynamically, and are thus not user-selectable options. Examples include the description, driver IDs, number of processors, and memory architecture fields, which are fixed according to the design of the hardware.

Note that except for the description, board designator, and driver ID fields, the second parameter in each field is the "restore-from-default" parameter and cannot be changed from the Hardware Selector or from Hardware Manager calls. In the Hardware Selector, if the user selects the Restore from Defaults button, this parameter is used to re-initialize the field. For an example and more detailed information, see the file excerpt and 1) under "Notes" in section 8.3 above, Hardware Setup File.

See below for explanations of the fields inside the Hardware Setup file.

Hardware Setup File Fields

Related Topics

Hardware Setup File Access

Selecting DSP/Analog Hardware Types

DSP/Analog Hardware Configuration

Hardware Values and Variables

General Topics

Hardware Manager Access

Hardware Selector

Hardware Setup File

Error Handling in Hardware Manager

Description Field in Hardware Setup File

[@j description field]

This field describes the hardware type. Usually, the name or model of the hardware is given, sometimes also including a mention or abbreviation of the type of DSP device on the hardware. The description field should be in text form, with no more than one (1) space separating words in the text, and should not exceed 50 characters.

Additional Hardware Setup File Fields

Board Designator

Driver IDs

Number of Processors

I/O Base Address

Memory Base Address

Bus Type

Analog Clock Type

Voltage Ranges

DSP Program File

Processor Clock

Memory Architecture

Board Designator Field in Hardware Setup File

[@j board designator field]

Entry of this field in the Hardware Setup file is essentially the same as its counterpart in the Hardware Selector. See Board Designator, above. Note that this field does not have a restore-from-default parameter. Invalid entry in this field, or entry which is not understood by a DSP Engine, will cause undefined results.

For a list of valid board designators, see either the Hardware Setup File (hwsetup.lst) or Signalogic DSP Software Reference Guide section 3.2, Board Designator, Driver ID, and Classification Table.

Although board designators should be selected by selecting an item in the Hardware Selector listbox, they can also be entered directly. New hardware types, with new board designators, can also be added to the Hardware Setup file. This might be necessary in a case where a designator or valid suffix is not listed or correctly listed in the Hardware Setup file. Board designators are usually divided into three parts:

1. manufacturer abbreviation
2. DSP chip type
3. "subspecies" suffix

For example, a fictitious designator might be 'WID88', designating the Widget Systems DSP-88 board, containing the Arkansas Instruments DSP8800 device. A 'WID88-B' designator might then indicate the PC-88 board, which contains the less expensive (but also less powerful) DSP8802 device. In the latter case, the '-B' is a suffix indicating "subspecies" hardware, which usually involves a board similar in DSP device type, onboard memory type and amount, and digital interface, but different in analog I/O capability.

Additional Hardware Setup File Fields

Description
Driver IDs
Number of Processors
I/O Base Address
Memory Base Address
Bus Type
Analog Clock Type
Voltage Ranges
DSP Program File
Processor Clock
Memory Architecture

Driver ID Fields in Hardware Setup File

[@j driver ID field]

The driver ID fields specify hardcoded constants which indicate the type of hardware to DSP Engine drivers for DSP/analog hardware. The first field is the major driver ID, which indicates a "main hardware type" value. Major driver ID values are grouped according to DSP chip family type. The second field is the minor driver ID, which, if non-zero, indicates a hardware type closely related to the major type, but different in analog I/O or host PC interface characteristics.

NOTE: Grouping major ID values by DSP chip family type for various DSP/analog hardware must be maintained at all times. The major ID value is used by Hardware Library functions such as DSGetWordLength, DSDSPToIEEE, and DSIEEEToDSP to determine the type of DSP chip being used.

For a list of driver ID values, see Signalogic DSP Software Reference Guide section 3.2, Board Designator, Driver ID, and Classification Table.

Additional Hardware Setup File Fields

Description

Board Designator

Number of Processors

I/O Base Address

Memory Base Address

Bus Type

Analog Clock Type

Voltage Ranges

DSP Program File

Processor Clock

Memory Architecture

Number of Processors Field in Hardware Setup File

[@j number of processors field]

This field indicates the number of processors on the hardware. Invalid entry in this field will cause undefined results.

Additional Hardware Setup File Fields

Description

Board Designator

Driver IDs

I/O Base Address

Memory Base Address

Bus Type

Analog Clock Type

Voltage Ranges

DSP Program File

Processor Clock

Memory Architecture

I/O Base Address Field in Hardware Setup File

[@j I/O base address field]

This field indicates the I/O base address, if applicable, for the hardware type. The restore-from-default half of this field should be left equal to the factory default setting for the hardware. See I/O Base Address, above, for more information.

Additional Hardware Setup File Fields

Description

Board Designator

Driver IDs

Number of Processors

Memory Base Address

Bus Type

Analog Clock Type

Voltage Ranges

DSP Program File

Processor Clock

Memory Architecture

Memory Base Address Field in Hardware Setup File

[@j memory base address field]

This field indicates the memory base address, if applicable, for the hardware type. The restore-from-default half of this field should be left equal to the factory default setting for the hardware. For more

information, see Memory Base Address in section 8.2.2 above, DSP/Analog Hardware Configuration.

Additional Hardware Setup File Fields

Description

Board Designator

Driver IDs

Number of Processors

I/O Base Address

Bus Type

Analog Clock Type

Voltage Ranges

DSP Program File

Processor Clock

Memory Architecture

Bus Type Field in Hardware Setup File

[@j bus type field]

This field specifies the host computer bus type used by the hardware, and should be set to a '1' for most DSP/analog hardware. A '1' value indicates that the bus interface on the hardware can handle 80286-type string move instructions at a minimum. Below are valid Bus Type values:

0	8086 instruction set only
1	80286 instruction set (string move instructions)
2-10	Reserved for Signalogic

Additional Hardware Setup File Fields

Description

Board Designator
Driver IDs
Number of Processors
I/O Base Address
Memory Base Address
Analog Clock Type
Voltage Ranges
DSP Program File
Processor Clock
Memory Architecture

Analog Clock Type Field in Hardware Setup File

[@j analog clock type field]

The Analog Clock Type field specifies the type of analog I/O clock (sampling frequency clock) used on the DSP/analog hardware. This field should be one of the following:

0	none
1	external
2	fixed, generated onboard or onchip
3	software programmable, generated onboard or onchip

Additional Hardware Setup File Fields

Description
Board Designator
Driver IDs
Number of Processors
I/O Base Address
Memory Base Address

Bus Type

Voltage Ranges

DSP Program File

Processor Clock

Memory Architecture

Voltage Ranges Field in Hardware Setup File

[@j voltage ranges field]

The Voltage Ranges field sets the analog input and output voltage ranges for the selected DSP/analog hardware type. Entry is essentially the same as in the Hardware Selector. For more information, see Voltage Ranges in section 8.2.2 above, DSP/Analog Hardware Configuration.

Additional Hardware Setup File Fields

Description

Board Designator

Driver IDs

Number of Processors

I/O Base Address

Memory Base Address

Bus Type

Analog Clock Type

DSP Program File

Processor Clock

Memory Architecture

DSP Program File Field in Hardware Setup File

[@j DSP program file field]

The DSP Program File field sets the default executable DSP file for DSPower series functions to download to DSP/analog hardware if an explicit entry is not given. Entry is essentially the same as in the Hardware Selector. For more information, see DSP Program File in section 8.2.2 above, DSP/Analog Hardware Configuration.

Additional Hardware Setup File Fields

Description

Board Designator

Driver IDs

Number of Processors

I/O Base Address

Memory Base Address

Bus Type

Analog Clock Type

Voltage Ranges

Processor Clock

Memory Architecture

Processor Clock Field in Hardware Setup File

[@j processor clock field]

The Processor Clock field sets the value of the onboard clock rate being supplied to the DSP device on the DSP/analog hardware. Entry is in MHz, and is essentially the same as in the Hardware Selector. For more information, see Processor Clock in section 8.2.2 above, DSP/Analog Hardware Configuration.

Additional Hardware Setup File Fields

Description

Board Designator

Driver IDs

Number of Processors

I/O Base Address

Memory Base Address

Bus Type

Analog Clock Type

Voltage Ranges

DSP Program File

Memory Architecture

Memory Architecture Field in Hardware Setup File

[@j memory architecture field]

The Memory Architecture field specifies the type of memory architecture access used by the DSP device on the DSP/analog hardware. This field applies to onboard memory that is external to the DSP device, and not necessarily to onchip memory. This field should be one of the following:

L	linear, both program and data in the same memory space
H	modified Harvard architecture, with program and data in separate memory spaces
V	modified Harvard architecture with program and data in separate memory spaces, and with data memory further subdivided into vector memory spaces

Additional Hardware Setup File Fields

Description

Board Designator

Driver IDs

Number of Processors

I/O Base Address

Memory Base Address

Bus Type

Analog Clock Type

Voltage Ranges

DSP Program File

Processor Clock

8.3.2 Hardware Setup File Access

Hardware Manager (hwmgr.dll) contains several functions which can be used to access Hardware Setup File information from C/C++ and MATLAB programs. Below is a brief summary of some of these functions. For detailed information, see Signalogic DSP Software Reference Guide section 2.2.2, Hardware Manager Functions.

The two most basic functions are the DSReadHWSetupFile and DSWriteHWSetupFile calls. The former causes Hardware Manager to allocate global memory and read the Hardware Setup File into this memory, and the latter causes Hardware Manager to write the current memory contents out to the Hardware Setup File. If not yet called, the DSReadHWSetupFile function is invoked be Hardware Manager functions which require DSP/analog hardware configuration data upon being called the first time, including the DSShowHardwareSelectorDlg call.

If a DSReadHWSetupFile call made from a calling application is successful, the return value is the global memory handle to a series of HWLISTENTRY entries containing DSP/analog hardware configuration data. The number of entries can be determined by calling DSGetHWMgrNumEntries. The global memory handle is created using the GMEM_SHARED attribute.

NOTE: It is not recommended that calling applications access HWLISTENTRY entries directly using the global memory handle

returned by DSReadHWSetupFile. Instead, the DSGetHWMgrEntry call should be used.

The DSGetHWMgrEntry function can be used by a calling application to obtain a DSP/analog hardware configuration data. This function requires a pointer to a HWLISTENTRY structure, and an index indicating which entry to retrieve. Indexes can be found by using DSGetHWMgrEntryIndex, which returns the index matching a board designator string. Likewise, the DSSetHWMgrEntry function can be used by a calling application to store DSP/analog hardware configuration data, presumably after modification or creation of a HWLISTENTRY structure. Note that in the case of DSSetHWMgrEntry, if the index parameter is equal to the total current number of entries, the entry is added rather than replacing an existing entry. The current number of entries can be obtained by using the DSGetHWMgrNumEntries function.

Other miscellaneous functions are also available. The DSGetHWMgrCurVal and DSSetHWMgrCurVal functions can be used to obtain and set the current Hardware Selector hardware type listbox selection. The DSGetHWMgrVar and DSSetHWMgrVar functions can be used to obtain and set Board Designators for a specified hardware variable (for more information, see section 8.2.3 above, Hardware Values and Variables). Finally, the DSShowHardwareSelectorDlg function will cause the Hardware Selector dialog box to be displayed (for more information, see section 8.2 above, Hardware Selector).

See Also

Related Topics

Hardware Setup File Access

Selecting DSP/Analog Hardware Types

DSP/Analog Hardware Configuration

Hardware Values and Variables

General Topics

Hardware Manager Access

Hardware Selector

Hardware Setup File

Error Handling in Hardware Manager

HWLISTENTRY Structure

[@j HWLISTENTRY structure, HWLISTENTRY, PHWLISTENTRY]

Below is a reconstruction of the HWLISTENTRY structure, which is used in Hardware Manager calls such as DSGetHWMgrEntry and DSSetHWMgrEntry. The reprint below is divided into the following sections:

- information that is maintained in the Hardware Setup File
- information that is maintained internally by Hardware Manager, and is not stored in the Hardware Setup File

Each element in the structure is commented with a brief description. See also the notes below, after the structure definition, and section 8.3.1 above, Hardware Setup File Fields.

typedef struct {

// type	**// element**	**// comment**
DESCSTR	szMfg;	// manufacturer field
DESCSTR	szDescription;	// description field: name/model, DSP type, etc.
DESCSTR	szBoardDesignator;	// board designator field
short int	MajorDriverID;	// major DSP/analog hardware driver ID field
short int	MinorDriverID;	// minor DSP/analog hardware driver ID field
char	szCodeFile[144];	// DSP program file field (current executable code filename)
char	szCodeFileDef[144];	// default DSP program file
VALSTR	szIOBaseAddr;	// current I/O base address field (in hex)
VALSTR	szIOBaseAddrDef;	// default I/O base address
VALSTR	szMemBaseAddr;	// current memory base address field (in hex)
VALSTR	szMemBaseAddrDef;	// default base memory address
short int	BusType;	// bus type field
short int	BusTypeDef;	// default bus type

VALSTR	szProcClock;	// processor clock field (specified in MHz)
VALSTR	szProcClockDef;	// default processor clock
VALSTR	szVoltageRanges;	// voltage ranges field (should be entered as 'input, output')
VALSTR	szVoltageRangesDef;	// default voltage ranges
short int	AnalClock;	// analog clock type field: 0 = external clock, 1 = fixed internal, 2 = prog internal
short int	AnalClockDef;	// default analog clock type
VALSTR	szMemArch;	// memory architecture field: "L" = linear, "H" = modified Harvard, "V" = vector
VALSTR	szMemArchDef;	// default memory architecture
short int	NumProcessors;	// number of processors field
short int	NumProcessorsDef;	// default number of processors

// following parameters not stored in Hardware Setup File

// type	**// element**	**// comment**
short int	CallClass;	// calls-allowed classification
short int	Flags;	// internal information used to mark pending updates, deletes, etc.

} HWLISTENTRY;

typedef HWLISTENTRY FAR* PHWLISTENTRY;

Notes

1) HWLISTENTRY structures should be completely initialized to zero before using, so default values can take effect for parameters not used. This can easily be accomplished using a call such as

```
_fmemset(&HWListEntry, 0, sizeof(HWLISTENTRY));
```

2) Supporting type definitions used above:

```
typedef char DESCSTR[50];
typedef char VALSTR[10];
```

Related Topics

Hardware Setup File Access

Selecting DSP/Analog Hardware Types

DSP/Analog Hardware Configuration

Hardware Values and Variables

General Topics

Hardware Manager Access

Hardware Selector

Hardware Setup File

Error Handling in Hardware Manager

8.4 Error Handling in Hardware Manager

The Hardware Manager error handling scheme is straightforward. Most calls return a non-zero value on success, and a zero (NULL) on failure. When calls into hwmgr.dll fail, there are two levels of error possibilities: the DLL itself (or "base level"), and engine-related. The DSGetHWMgrErrorStatus and DSGetEngineErrorStatus calls can be used to attempt to learn more information about the most recent failure.

Note that the DSGetEngineErrorStatus call is actually inside enmgr.dll. Note also that Engine Manager maintains a "status listbox" which can be made visible using the DSShowEngMgrStatusWindow call. This status window shows a history of *both status and error* text displays resulting from most Engine Manager and Hardware Manager calls. For complete information on error handling in Engine Manager, see section 7.5, Error Handling in Engine Manager.

Related Topics

General Topics

Hardware Manager Access

Hardware Selector

Selecting DSP/Analog Hardware Types

DSP/Analog Hardware Configuration

8.4.1 DLL Error Handling in Hardware Manager

Hardware Manager has a "base level" error/status routine which can be used to look for general DLL error conditions not necessarily associated with engine problems. Below is an example showing further investigation after a DSSetHWMgrCurVal:

```
if (!DSSetHWMgrCurVal(szBoardDesignator)) {

    nErrorType = DSGetHWMgrErrorStatus();
    ...do error processing...
}
```

In the above example, the actual return value of DSSetHWMgrCurVal would be NULL in the case of failure. Thus, to find out more information, the DSGetHWMgrErrorStatus call is used.

Below is a partial list of error values returned by DSGetHWMgrErrorStatus, along with a brief description of their meaning. These constants can be found mostly in the hwmgr.h file and partly in the dllerr.h file; for a complete list all DSPower series software error conditions by error number, see Error and Status Condition Numeric List and Cross-Reference, in Signalogic DSP Software Reference Guide chapter 7, Error and Status Conditions.

dllerr.h or hwmgr.h constant	Summary of Meaning
DSCOULDNOTINITIALIZEDLL	Hwmgr.dll itself could not be initialized.
DSUNABLETOREADHWSETUPFILE	Unable to read the Hardware Setup File.
DSCOULDNOTLOCKHWLIST	Unable to lock the global memory hardware list maintained inside hwmgr.dll
DSBOARDDESIGNATORNOTLISTED	A board designator input to a hwmgr.dll function was not

	listed in the Hardware Setup File.
DSCOULDNOTFINDHWSETUPFILE	The Hardware Setup File could not be located.
DSCOULDNOTINCREASEHWLISTMEM	The global memory hardware list maintained inside hwmgr.dll could not be increased in size (to accommodate additional Hardware Setup File entries).

Note that in C/C++ programs, it is safest to treat the above error definitions as integer types, since some values are "positive" and some "negative". See Error and Status Condition Numeric List and Cross Reference, in Signalogic DSP Software Reference Guide chapter 7, Error and Status Conditions, for more information about this grouping.

General Topics

Hardware Manager Access

Hardware Selector

Selecting DSP/Analog Hardware Types

DSP/Analog Hardware Configuration

Hardware Values and Variables

Hardware Setup File

Error Handling in Hardware Manager